15 | 常见的性能问题之CPU篇：如何让CPU的运行不受阻碍?

2019-12-30 庄振运

讲述：庄振运
时长 13:57 大小 12.79M

你好，我是庄振运。

从这一讲开始，我们探讨分析几个最容易发生性能问题的领域：CPU、内存、存储和网络。

这一讲先来讨论关于 CPU 的常见性能问题。首先我们从硬件的角度，来看看 CPU 的性能取决于哪些因素，然后分析一下 CPU 的内部结构。接着我们探讨和 CPU 性能相关的软件系统，看看 CPU 运行时侯的调度和切换。

CPU 的性能决定因素

宏观来讲，一台服务器里面的 CPU 性能取决于好几个因素，包括有多少处理器、多少个核、时钟主频是多少、有没有 Turbo 模式、处理器内部的运算架构以及和 CPU 紧密交互的其他部件的性能。

CPU 的更新换代很频繁，基本上每两年就会更新一代。比如 Intel 的 CPU，最近 10 年已经经历了 5 代左右。每一代都有主频的变化，而且有好几个变种。

下面的表格描述了从十年前（也就是 2009 年）的 SandyBridge，到后来的 IvyBridge、Haswell、Broadwell，直到 Skylake。注意，对后面的三代，我分别列出了其中的两种变化——单处理器（1P）和双处理器（2P）。

	SandyBridge	IvyBridge	Haswell (1P)	Haswell (2P)	Broadwell (1P)	Broadwell (2P)	Skylake (1P)	Skylake (2P)
Cores (per socket)	8	10	12	12	16	14	18	20
主频 Frequency	2.6GHz	2.8GHz	2.5GHz	2.5GHz	1.8GHz	2.4GHz	1.6GHz	2.0GHz
Turbo	2.9GHz	3.1GHz	2.9GHz	2.9GHz	2.3GHz	2.9GHz	2.2 GHz	2.7GHz

大体上我们可以看出，虽然 CPU 更新换代，但是处理器的时钟主频基本不再提高，甚至变得更低了。这样的目的是降低 CPU 的功耗。比如 SandyBridge 的时钟频率是 2.6GHz，但是到了 Skylake，反而降低到了 2GHz。

为了提升单个处理器的性能，每个处理器里面的核数却越来越多，这样就可以尽量的提升并行处理能力。比如 SandyBridge 的每个处理器只有 8 个核，而 Skylake 则多达 20 个核。

而且我们也看到，每一代 CPU 都允许 Turbo 模式，就是让 CPU 的主频提高。目的是可以让处理器在特殊情况下，用提高功耗的代价来增加主频，从而获得更高性能。

CPU 的内部结构

CPU 的性能也取决于它的内部结构设计。很多程序员对 CPU 的内部机构不是完全清楚，尤其是对相关的术语之间的区别和联系一知半解，比如多处理器和多核、逻辑 CPU 和硬件线程、超线程，以及 L1/L2/L3 三级缓存等。

之所以对这些结构不甚了解，主要原因是现代处理器变得复杂，普遍采用多处理器，多核以及内部的各种优化处理来提高 CPU 性能。我们今天就从外到内，从宏观到微观地介绍一

下。

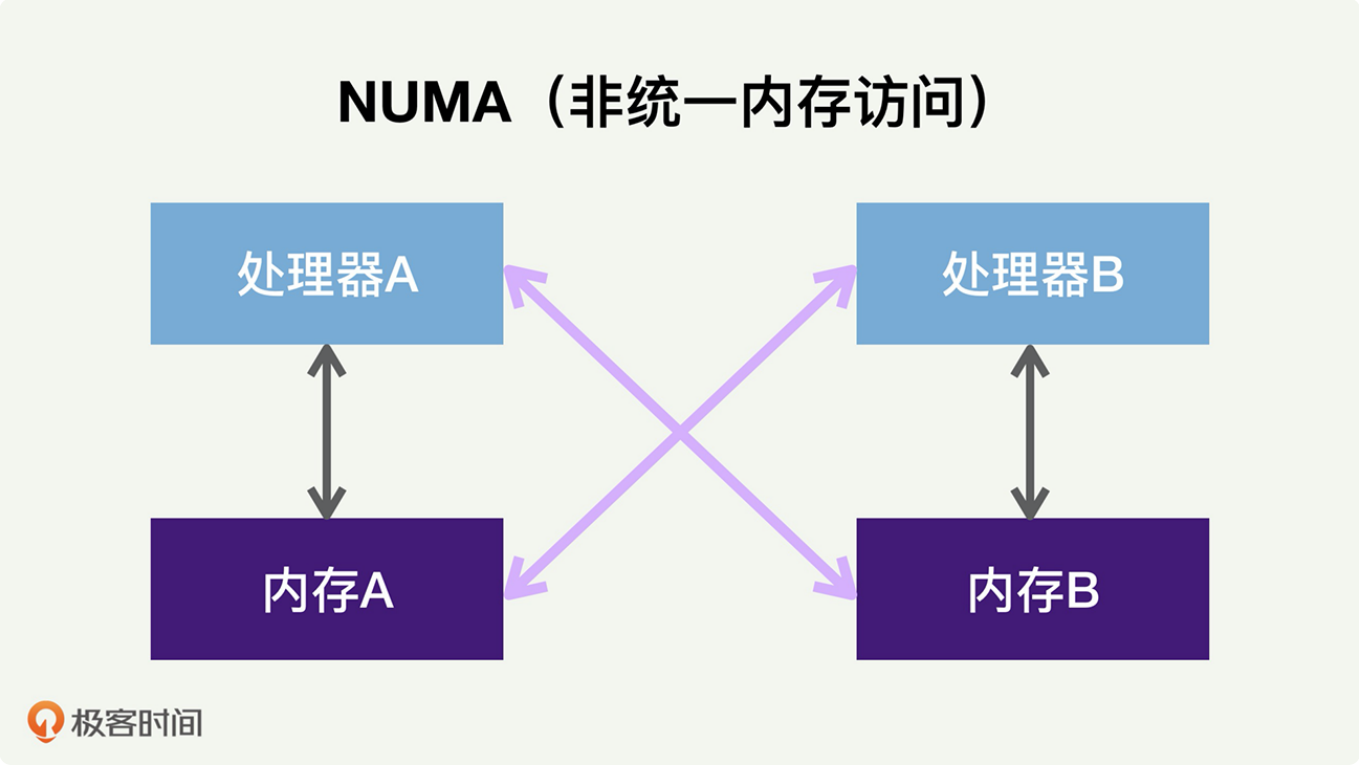
我注意到，这方面的很多中文术语，大家有时候用法不一致，所以很容易混淆。为了清楚描述，你尤其要注意一下我用的术语（包括英文）。

多处理器和 NUMA

现在的 CPU 普遍采用多处理器（Socket）来提高 CPU 性能，每个处理器都有自己可以直接访问的本地内存（Local Memory）。一般来讲，这里面每个处理器的性能和内存大小都是一样的。每个处理器也都可以访问其他处理器的内存，这些内存就相当于是外地 / 远程内存（Remote Memory）。

当 CPU 处理器访问本地内存时，会有较短的响应时间（称为本地访问 Local Access）。而如果需要访问外地 / 远程内存时候，就需要通过互联通道访问，响应时间就相比本地内存变慢了（称为远端访问 Remote Access）。所以 NUMA（Non-Uniform Memory Access）就此得名。

下图展示了两个处理器的 NUMA 架构。



如果处理器 A 访问内存 A，就是本地访问。如果它访问内存 B，就是远端访问，内存的访问延迟大大增加。

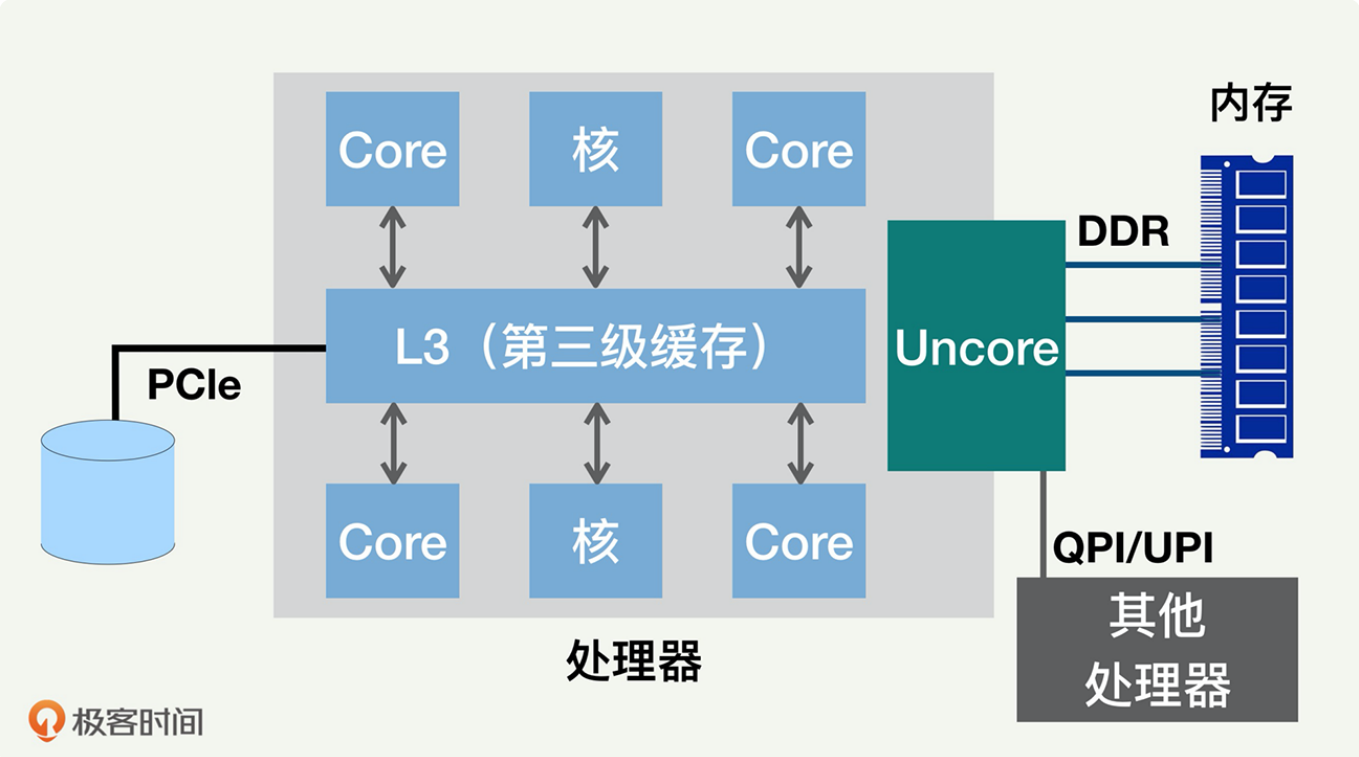
采用多处理器和 NUMA 架构的主要原因，是提高整个 CPU 的并行处理性能。每一台服务器可以同时运行很多程序和进程。对每一个进程和线程而言，当它运行在某一个处理器上时，它所对应的内存使用默认的分配方案是——优先尝试在请求线程当前所处的处理器的本地内存上分配。如果本地内存不足，才会分配到外地 / 远程内存上去。

多核结构和多级缓存

我们再看看每个处理器内部的结构。我们刚刚讲到的处理器，内部一般都是多核（Core）架构。随着多核处理器的发展，CPU 的缓存通常分成了三个级别：L1、L2 和 L3。

级别越小就越接近 CPU，速度更快，同时容量也越小。L1 和 L2 一般在核的内部，我们下一讲还会详细讲。L3 缓存是三级缓存中最大的一级，同时也是最慢的一级；在同一个处理器内部的核会共享同一个 L3 缓存。

除了多个核以及 L3 缓存外，处理器上一般还有非核心处理器（Uncore），里面含有和指令运行不直接相关的组件，包括 QPI 控制器和存储器一致性监测组件，如下图所示。



超线程（Hyperthreading，HT）

一个核还可以进一步分成几个逻辑核，来执行多个控制流程，这样可以进一步提高并行程度，这一技术就叫超线程，有时叫做 simultaneous multi-threading（SMT）。

超线程技术主要的出发点是，当处理器在运行一个线程，执行指令代码时，很多时候处理器并不会使用到全部的计算能力，部分计算能力就会处于空闲状态。而超线程技术就是通过多线程来进一步“压榨”处理器。

举个例子，如果一个线程运行过程中，必须要等到一些数据加载到缓存中以后才能继续执行，此时 CPU 就可以切换到另一个线程，去执行其他指令，而不用去处于空闲状态，等待当前线程的数据加载完毕。

通常，一个传统的处理器在线程之间切换，可能需要几万个时钟周期。而一个具有 HT 超线程技术的处理器只需要 1 个时钟周期。因此就大大减小了线程之间切换的成本，从而最大限度地让处理器满负荷运转。

一个核分成几个超线程呢?

这个数字会根据 CPU 架构有所变化；Intel 一般是把一个核分成 2 个。

“这台计算机有多少 CPU？”

我们经常会问这个问题，结合我们刚刚讲的知识，就很容易回答了。

比如，如果一台计算机有两个处理器，每个处理器有 12 个核，而且采用了 HT 超线程，那么总的 CPU 数目就是 48，就是 2×12×2。这个数字 48，就是我们平时用监控软件和命令看到的 CPU 的数量。比如，Linux 的 top 或者 vmstat 命令，显示的 CPU 个数就是这样算出来的。

CPU 性能指标和常见性能问题

我们继续探讨 CPU 的性能指标和常见性能问题，这方面很多资料都有涉及，我们提纲挈领地总结一下。

最表层的 CPU 性能指标，就是 CPU 的负载情况和使用率。CPU 使用率又进一步分成系统 CPU、用户 CPU、IO 等待 CPU 等几个指标。你执行一下 top 命令就会看到。

需要注意的是，因为 CPU 架构的复杂性，以及和其他部件的交互，CPU 的使用率和负载的关系往往不是线性的。

也就是说，如果 10% 的 CPU 使用率可以每秒处理 1 千个请求，那么 80% 的 CPU 使用率能够处理多少请求呢？不太可能处理每秒 8 千个请求，往往会远远小于这个数字。

衡量一个应用程序对 CPU 使用效率时，往往会考察 CPI（Cycles Per Instruction，每指令的周期数）和 IPC（Instructions Per Cycle，每周期的指令数）。这两个指标有助于识别运行效率高或低的应用程序。而一台计算机的 CPU 性能一般用 MIPS（Millions of Instructions Per Second）来衡量，表示每秒能运行多少个百万指令，MIPS 越高，性能越高。MIPS 的计算很简单，就是时钟频率×IPC。

继续往深处分析，CPU 常见的各种中断包括软中断和硬中断。除此之外，还有一种特殊的中断：上下文切换。这些指标需要和每个核挂钩，理想情况下是各个核上的中断能够均衡。如果数量不均衡，往往会造成严重的性能问题——有的核会超载而导致系统响应缓慢，但是其他的核反而空闲。

和 CPU 相关的性能问题，基本上就是表现为 CPU 超载或者空闲。

如果是 CPU 超载，那么就要分析为什么超载。多数情况下都不一定是合理的超载，比如说多核之间的负载没有平衡好，或者 CPU 干了很多没用的活，或者应用程序本身的设计需要优化等等。反之，如果是 CPU 空闲，那就需要了解为什么空闲，或许是指令里面太多内存数据操作，从而造成 CPU 停顿，也或许是太多的分支预测错误等，这就需要具体分析和对症下药的优化。

CPU 对多线程的执行顺序是谁定的呢？

是由内核的进程调度来决定的。内核进程调度负责管理和分配 CPU 资源，合理决定哪个进程该使用 CPU，哪个进程该等待。进程调度给不同的线程和任务分配了不同的优先级，优先级最高的是硬件中断，其次是内核（系统）进程，最后是用户进程。每个逻辑 CPU 都维护着一个可运行队列，用来存放可运行的线程来调度。

CPU 的性能监测工具

我们最后讲一下 CPU 性能监测方面的工具。和 CPU 监测相关的工具挺多的，而且往往每个工具都包含很多不同的有用的信息。

比如在 Linux 上，最常用的 Top 系统进程监控命令。Top 是一个万金油的工具，可以显示出 CPU 的使用、内存的使用、交换内存和缓存大小、缓冲区大小、各个进程信息等。

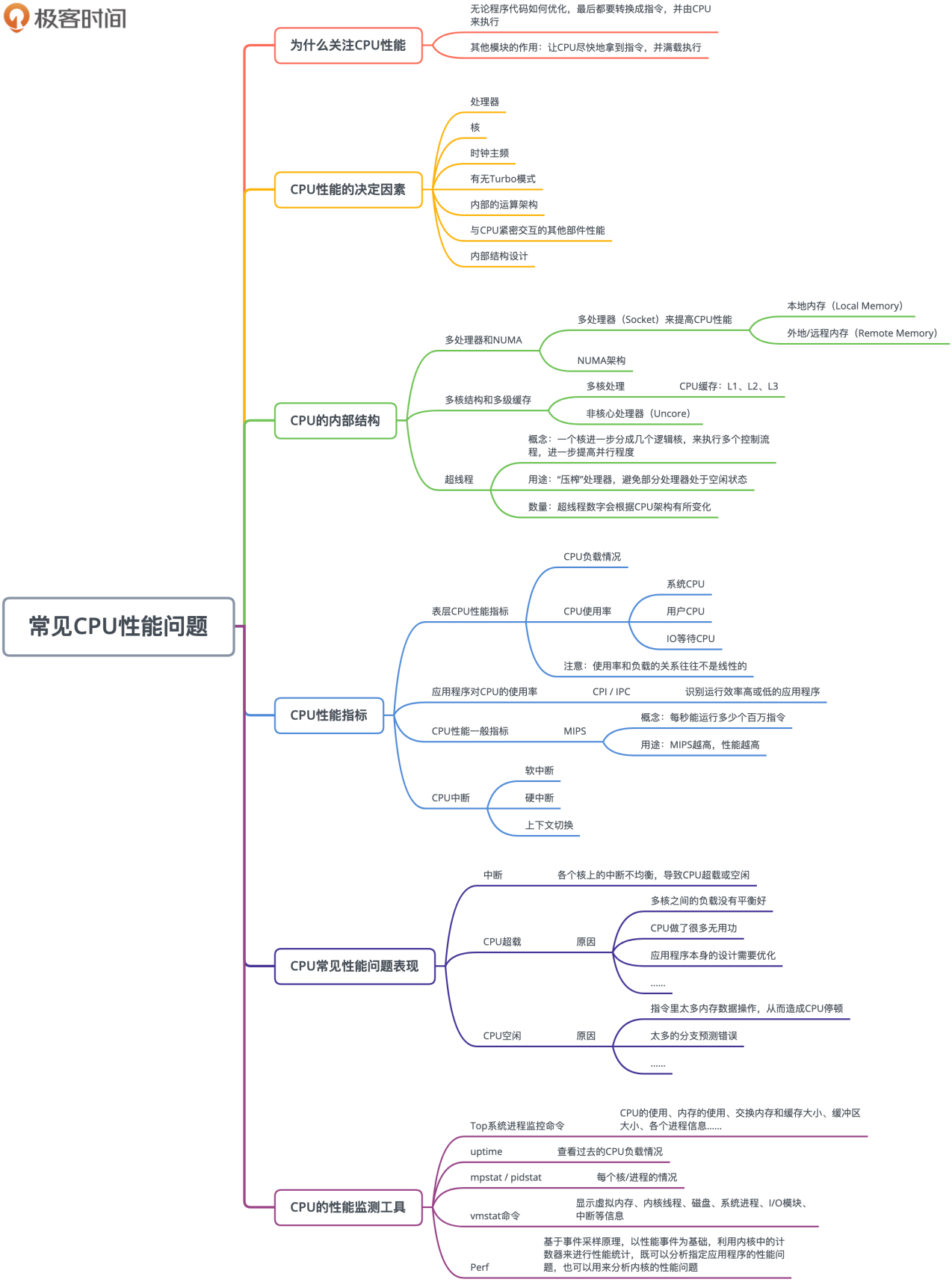
如果想要查看过去的 CPU 负载情况，可以用 uptime。也可以用 mpstat 和 pidstat，来分别查看每个核还有每个进程的情况。另一个常用的 vmstat 命令可以用于显示虚拟内存、内核线程、磁盘、系统进程、I/O 模块、中断等信息。

对有经验的性能工程师来讲，有一个类似于“瑞士军刀”一样的好工具：Perf。

Perf 是 Linux 上的性能剖析（profiling）工具，极为有用。它是基于事件采样原理，以性能事件为基础，利用内核中的计数器来进行性能统计。它不但可以分析指定应用程序的性能问题，也可以用来分析内核的性能问题。

总结

这一讲我们讨论了计算机的运算核心，CPU 的结构，尤其是它内部的和性能相关的部件，并澄清了一些术语。



CPU 是服务器性能的最重要的部分；因为不管程序代码如何优化，最后都要转换成指令，让 CPU 来执行。不管其他部件如何和 CPU 交互，最终目的是让 CPU 尽快地拿到指令，并满载执行。

唐代有个诗人叫李贺，他曾经形容跨马奔驰的愿景：

“大漠沙如雪，燕山月似钩。

何当金络脑，快走踏清秋。”

我们常把 CPU 类比大脑，CPU 性能优化的目标，就是让它的运行不受阻碍，如千里马一样任意驰骋。

现代 CPU 提升性能的主要途径是并行化，这方面的策略包括：多处理器、多核、超线程，另外还有流水线架构和超标量等等，都是为了提高并行处理能力。

思考题

你工作中一定碰到过 CPU 方面的性能问题吧？总结一下，有几种表现形式？

> 比如是总体 CPU 使用量太高，还是计算机的几个核负载不均衡？负载不均衡是什么原因导致的呢？是线程数不够，还是系统调度的问题？

欢迎你在留言区分享自己的思考，与我和其他同学一起讨论，也欢迎你把文章分享给自己的朋友。

© 版权归极客邦科技所有，未经许可不得传播售卖。 页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇　14 | 性能分析概述：性能问题归根结底是什么原因?

精选留言 (3)

写留言

Di Yu
2019-12-30

请问老师，用什么工具才能测出来CPU几个核负载不均衡呢?

2

Scott
2019-12-30

我遇到过一个sys cpu高的问题，现象就是sys cpu至少10%，一重启就好了，最后发现是cgroup泄漏的问题，这个问题排查过程真是非常艰难。

展开

Linuxer
2019-12-30

请问老师后续有与实验结合的讲解吗？我觉得大多数人都知道概念和监测的方法，但是都不知道如何分析出具体的深层次的原因？比如NUMA里面跨节点内存访问对性能到底有多

大影响呢？比如流水线停顿对性能的影响又是如何表现的等等

展开